Amendment to the Claims:

This listing of claims will replace all prior versions, and listings of claims in the application:

Listing of Claims

1   1-31. (Canceled)

1   32.   (Currently amended): A method of transforming data, the method

2 comprising:

3   positioning a definition pointer to point at a first compound transform definition

4 within a transform definition file;

5   invoking a first parallel processing thread to read the pointed at first compound

6 transform definition;

7   searching data to be transformed for a data element to be transformed, the search

8 being responsive to the first compound transform definition;

9   calling a dynamic function defined in the transform definition file, the dynamic

10 function located elsewhere in the transform definition file from the definition pointer position;

11   transforming any found data element into an output data file, responsive to the

12 first compound transform definition and called dynamic function, a data structure of the output

13 data file being responsive to a data structure of the first compound transform definition;

14   determining a type of the read first compound transform definition;

15   based on a determination that the first compound transform definition is

16 compound, processing each sub-definition of the read first compound transform definition by

17 repeating the positioning, invoking, searching, calling, and transforming for each sub-definition

18 in order to transform each sub-definition recursively, wherein the data elements transformation

19 includes nesting of a data element;

20   determining if all sub-definitions of the first compound transform definition have

21 been processed;

22   positioning a definition pointer to point at a second compound transform

23   definition within the transform definition file;

24   invoking a second parallel processing thread to read the pointed at second

25   compound transform definition;

26   searching data to be transformed for another data element to be transformed, the

27   search being responsive to the second compound transform definition;

28   transforming any found data element into the output data file, responsive to the

29   second compound transform definition, the data structure of the output data file being responsive

30   to the data structure of the second compound transform definition; and

31   if no data element is found to be transformed, adding one or more output data

32   elements to the output data file responsive to the read first compound transform definition, the

33   data to be transformed having no contribution to the output data element.

1   33-36. (Canceled)

1   37.   (Previously presented): The method of claim 32, wherein the read first

2   compound transform definition includes a value parameter configured to specify a value for

3   inclusion in the output data file.

1   38.   (Original): The method of claim 32, wherein the data element is a

2   compound data element and the read transform definition includes a source record parameter

3   configured to specify the compound data element.

1   39.   (Previously presented): The method of claim 32, wherein the read first

2   compound transform definition is in a meta-language format.

1   40.   (Original): The method of claim 32, wherein the data to be transformed

2   data is in a meta-language data format.

OID-2005-309-01

1   41.   (Previously presented): The method of claim 32, wherein the read first

2   compound transform definition includes a transform element having an output field name and a

3   source field parameter.

1   42.   (Previously presented): The method of claim 32, wherein the read first

2   compound transform definition includes a value parameter configured to populate a field in the

3   output data file.

1   43.   (Canceled)

1   44.   (Currently amended): A method of transforming data, the method

2   comprising:

3   positioning a definition pointer to point at a first compound transform definition,

4   the first compound transform definition being within a transform definition file;

5   invoking a first parallel processing thread to read the pointed at first compound

6   transform definition and sub-definitions of the first compound transform definition;

7   positioning a first payload pointer to point at a first data element to be

8   transformed, the positioning of the first payload pointer being responsive to a data structure of

9   the first compound transform definition;

10   calling a dynamic function defined in the transform definition file, the dynamic

11   function located elsewhere in the transform definition file from the definition pointer position;

12   transforming the first data element into an output data file, responsive to the read

13   first compound transform definition and called dynamic function;

14   determining a type of the read first compound transform definition;

15   based on a determination that the first compound transform definition is

16   compound, processing each sub-definition of the read first compound transform definition by

17   repeating the positioning, invoking, positioning, calling, and transforming for each sub-definition

18   in order to transform each sub-definition recursively, wherein the data elements transformation

19   includes nesting of a data element;

20   determining if all sub-definitions of the first compound transform definition have

21   been processed;

22   positioning the definition pointer to point at a second compound transform

23   definition, the second compound transform definition being within the transform definition file;

24   invoking a second parallel processing thread to read the pointed at second

25   compound transform definition and sub-definitions of the second compound transform

26   definition;

27   positioning a second payload pointer to point at a second data element to be

28   transformed, the positioning of the second payload pointer being responsive to a data structure of

29   the second compound transform definition;

30   transforming the second data element into the output data file, responsive to the

31   read second compound transform definition; and

32   if no first or second data element is found to be transformed, adding one or more

33   output data elements to the output data file responsive to the read first compound transform

34   definition, the data to be transformed having no contribution to the output data element.

1   45-47. (Canceled)

1   48.   (Original): The method of claim 44, further including determining if all

2   sub-elements of a compound element have been transformed and, if the determination returns a

3   value of YES, returning to a calling process.

1   49.   (Canceled)

1   50.   (Previously presented): The method of claim 44, further including un-

2   nesting the data element to be transformed.

1   51.   (Previously presented): The method of claim 44, wherein the read first

2   compound transform definition includes a source field parameter configured to specify the data

3   element.

OID-2005-309-01

1   52.   (Previously presented): The method of claim 44, wherein the read first

2   compound transform definition includes a source record parameter configured to specify the

3   compound data element.

1   53-55. (Canceled)

1   56.   (Previously presented): The method of claim 44, wherein the transform

2   definition file includes a tree data structure.

1   57-58. (Canceled)

1   59.   (Currently Amended): A computer readable storage media having

2   embodied thereon data, the data comprising:

3   computer instructions configured to position a definition pointer to point at a first

4   compound transform definition, the first compound transform definition being within a transform

5   definition file;

6   computer instructions configured to invoke a first parallel processing thread to

7   read the pointed at first compound transform definition and sub-definitions of the first compound

8   transform definition;

9   computer instructions configured to position a first payload pointer to point at a

10 first data element to be transformed, the positioning being responsive to a data structure of the

11 first compound transform definition;

12   computer instructions configured to call a dynamic function defined in the

13 transform definition file, the dynamic function located elsewhere in the transform definition file

14 from the definition pointer position;

15   computer instructions configured to transform the first data element into an output

16 data file, responsive to the read first compound transform definition and called dynamic function;

17   computer instructions configured to determine a type of the read first compound

18 transform definition;

OID-2005-309-01

19   based on a determination that the first compound transform definition is

20   compound, computer instructions configured to process each ~~sub- definition~~ sub-definition of the

21   read first compound transform definition by repeating the positioning, invoking, positioning,

22   calling, and transforming for each sub-definition in order to transform [[the]] each sub-definition

23   recursively, wherein the data elements transformation includes nesting;

24   computer instructions configured to determine if all sub-definitions of the first

25   compound transform definition have been processed;

26   computer instructions configured to position a second payload pointer to point at a

27   second data element to be transformed, the positioning being responsive to a data structure of the

28   second compound transform definition;

29   computer instructions configured to invoke a second parallel processing thread to

30   read the pointed at second compound transform definition and sub-definitions of the second

31   compound transform definition; [[and]]

32   computer instructions configured to transform the second data element into the

33   output data file, responsive to the read second compound transform definition; and

34   if no first or second data element is found to be transformed, computer

35   instructions configured to add one or more output data elements to the output data file responsive

36   to the read first compound transform definition, the data to be transformed having no

37   contribution to the output data element.

1   60-64. (Canceled)

1   65.   (Currently amended): An application system comprising:

2   a computing device comprising:

3   a memory; and

4   at least one processor operatively coupled to the memory, the at least one

5   processor configured to:

6   position a definition pointer to point at a first compound transform definition

7   within a transform definition file;

8   invoke a first parallel processing thread to read the pointed at first compound

9   transform definition;

10   search data to be transformed for a data element to be transformed, the search

11   being responsive to the first compound transform definition;

12   call a dynamic function defined in the transform definition file, the dynamic

13   function located elsewhere in the transform definition file from the definition pointer position;

14   transform any found data element into an output data file, responsive to the first

15   compound transform definition and called dynamic function, a data structure of the output data

16   file being responsive to a data structure of the first compound transform definition;

17   determine a type of the read first compound transform definition;

18   based a on determination that the first compound transform definition is

19   compound, process each sub-definition of the read first compound transform definition by

20   repeating the positioning, invoking, searching, calling, and transforming for each sub-definition

21   in order to transform the each sub-definition recursively, wherein the data elements

22   transformation includes nesting of a data element;

23   determine if all sub-definitions of the first compound transform definition have

24   been processed;

25   position a definition pointer to point at a second compound transform definition

26   within the transform definition file;

27   invoke a second parallel processing thread to read the pointed at second

28   compound transform definition;

29   search data to be transformed for another data element to be transformed, the

30   search being responsive to the second compound transform definition;

31   transform any found data element into the output data file, responsive to the

32   second compound transform definition, the data structure of the output data file being responsive

33   to the data structure of the second compound transform definition; and

34   if no data element is found to be transformed, add one or more output data

35   elements to the output data file responsive to the read first compound transform definition, the

36   data to be transformed having no contribution to the output data element.

OID-2005-309-01

1   66.   (Previously presented): The application system of claim 65, wherein the

2   at least one processor is further configured to select the transform definition file from a set of

3   transform definitions files, responsive to data associated with the data to be transformed.

1   67.   (Canceled)

1   68.   (Previously presented): The application system of claim 65, wherein the

2   at least one processor is further configured to add data to the output data file, the added data

3   being configured responsive to the transform definition file and having no contribution from the

4   data to be transformed.